



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/656,846

09/04/2003

Jack Chu

85804-020301

8387

(017887-0101)

32361

7590

10/16/2006

GREENBERG TRAURIG, LLP  
MET LIFE BUILDING  
200 PARK AVENUE  
NEW YORK, NY 10166

EXAMINER

WATT, CHRIS A

ART UNIT

PAPER NUMBER

2174

DATE MAILED: 10/16/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application No.

10/656,846

Applicant(s)

CHU ET AL.

Examiner

Chris Watt

Art Unit

2174

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 04 September 2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-41 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-41 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 28 June 2004 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)                        | 4) <input type="checkbox"/> Interview Summary (PTO-413)                     |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)               | Paper No(s)/Mail Date. _____  |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| Paper No(s)/Mail Date <u>6/28/2004</u> .   | 6) <input type="checkbox"/> Other: _____                                    |

## DETAILED ACTION

### *Claim Objections*

Claims 2 and 8 are objected to because of the following informalities:

a) claim 2: the phrase "in an opening a direction" should be changed to -- in an opening direction --

b) claim 8: the phrase "sheet present a ticker" should be changed to -- sheet presents a ticker --

Appropriate correction is required.

### *Claim Rejections - 35 USC § 102*

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless --

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claims 1-41 are rejected under 35 U.S.C. 102(b) as being anticipated by Pasquali ("Pasquali", US #6,272,493).

As per independent claim 1, Pasquali teaches a browser user interface (fig. 1a; col. 5, line 52; *browser client*), wherein a browser executes on a client system (fig. 1c; col. 7, line 2; *client system*) coupled to a server (fig. 1a; col. 7, lines 18-19; *may be coupled with SVR system 102*) over a network (fig. 1a; col. 7 line 19; *via the electronic data network*) and provides a browser user interface (fig. 3; col. 11 line 15; *browser client*) to server resources (fig. 3, step S3-2; col. 11 lines 15-16; *all files and functions*

Art Unit: 2174

*and content stored with in [sic] server side systems)* and wherein the browser includes functionality to request pages from servers over the network (fig. 3; col. 11 line 17; *after requesting the same via an electronic data network*) and to process received pages (fig. 3a, step S3-3; col. 11 lines 19-20; *constructs a web site ... based on the received content*) for presentation to a browser user (fig. 3a, step S3-4; col. 11 lines 26-27; *manifest the web site view*), the browser user interface comprising:

a graphical display for presenting presented portions of browser pages to the user (col. 6 lines 11-12; *dynamic window in which WWW content is normally displayed*);

a user input device for accepting user input related to a page displayed in the browser (col. 6 lines 38-40; *control objects ... react to events (e.g., mouse clicks, mouse-overs, double-clicks, etc.)*); and

storage for dynamic interface elements received by the browser in connection with received pages (fig. 1c, *data storage sub system 114*; col. 7 lines 7-8; *access and download HTML documents ... and other related files*), wherein a dynamic interface element is able to be presented and modified in response to selected user input without requiring further interaction with a server (col.4 lines 36-41; *window object(s) ... may be updated and loaded with content ... without requiring the content manifestation environment to be refreshed*).

As per claim 2, which is dependent on claim 1, Pasquali teaches a browser user interface wherein the dynamic interface elements include slide sheets (fig. 2a; *content display section 242*; col. 9 line 62; *may either be minimized or maximized*),

Art Unit: 2174

wherein a slide sheet opens in an opening a direction on the display in response to the selected user input (fig. 2a; col. 8 lines 41-42; i.e., *via the up-arrow icon*).

As per claim 3, which is dependent on claim 2, Pasquali teaches a browser user interface wherein a slide sheet includes a plurality of tabs (col. 9 line 56; *window modules 202 arranged in table fashion*) each representing a subset of the browser user interface elements (col. 9 line 49; *content management environment (CME)*) presented on the slide sheet and logic for switching among tabs in response to user input for presenting corresponding browser user interface elements (col. 9, line 67- col. 10, line 1; *module control icons (MCs) which correspond to associated control logic*) without requiring further interaction with a server (col.4 lines 36-41; *window object(s) ... may be updated and loaded with content ... without requiring the content manifestation environment to be refreshed*).

As per claim 4, which is dependent on claim 3, Pasquali teaches a browser user interface wherein a slide sheet includes a scroll user interface elements for scrolling user interface elements of the slide sheet within a display area smaller than the presentation of all user interface elements of the slide sheet (fig. 1d, *scroll controls 126, scroll bar 128*; col. 9 lines 10-11; *when content extends beyond the ... size of content display section 122*).

As per claim 5, which is dependent on claim 2, Pasquali teaches a browser user interface further comprising the user interface elements for allowing the user to resize the slide sheet in one or more directions (col. 47, line 37; *A user may adjust window sizes*) without requiring further interaction with the server (col.4 lines 36-

41; *window object(s) ... may be updated and loaded with content ... without requiring the content manifestation environment to be refreshed*).

As per claim 6, which is dependent on claim 2, Pasquali teaches a browser user interface wherein the slide sheet is semi-transparent (col. 33 line 30; *spacer graphic ie. a transparent image that is set by the width & height values specified*).

As per claim 7, which is dependent on claim 2, Pasquali teaches a browser user interface wherein the slide sheet is associated with a topic specific page (col. 1, lines 56-57; *may select topic areas*), wherein topics include news, sports, weather, commentary, commerce, music, movies, games or local information (fig. 2a, *NEWS*; col. 1 line 21; *news, weather, sports*).

As per claim 8, which is dependent on claim 7, Pasquali teaches a browser user interface wherein the slide sheet is associated with a finance page (col. 2, line 57; *financial feeds*) and the slide sheet present a ticker lookup interface (col. 3, line 40; *updated stock tickers*).

As per claim 9, which is dependent on claim 1, Pasquali teaches a browser user interface further comprising:

logic to track user interface interactions with the dynamic interface elements (col. 17 lines 13-15; *GUI manipulation of window module attributes stored*);  
and

logic for generating messages to a server corresponding with the dynamic interface elements, wherein a message to the server indicates user interactions (col. 17,

Art Unit: 2174

lines 11-12; *database table is generated to store window module properties and the like*) and wherein such interactions are effected independent of whether the server receives the message (col. 18, lines 12-13; *may be downloaded ... within their WWW browser clients*).

As per claim 10, which is dependent on claim 1, Pasquali teaches a browser user interface wherein pages with dynamic interface elements received from a server include substitute presentations for presentation by browsers that do not support dynamic interface elements (col. 10, lines 46-48; *may be used to facilitate manifestation of content that would otherwise not be destined for window module manifestation*).

As per claim 11, which is dependent on claim 1, Pasquali teaches a browser user interface wherein the dynamic interface elements include a menu bar, wherein the menu bar is a user interface elements for providing navigation among a hierarchy of selection stored (col. 9, lines 56-57; *web site (environment) controls*; see also col. 51, lines 8-11; *initiates a pop-up window ... "popMenu()"*) as such that a user can navigate within the menu bar without requiring further interaction with the server (col. 4 lines 36-41; *window object(s) ... may be updated and loaded with content ... without requiring the content manifestation environment to be refreshed*).

As per claim 12, which is dependent on claim 11, Pasquali teaches a browser user interface wherein the browser further comprises:

logic to track user interface interactions with the menu bar (col. 17 lines 13-15; *GUI manipulation of window module attributes stored*); and

logic for generating messages to a server corresponding with the menu bar, wherein a message to the server indicates user interactions with the menu bar (col. 17, lines 11-12; *database table is generated to store window module properties and the like*) and wherein such interactions are effected independent of whether the server receives the message (col. 18, lines 12-13; *may be downloaded ... within their WWW browser clients*).

As per claim 13, which is dependent on claim 1, Pasquali teaches a browser user interface further comprising a cache for caching dynamic interface elements at the client (col.4 lines 36-41; *window object(s) ... may be updated and loaded with content ... without requiring the content manifestation environment to be refreshed*).

As per independent claim 14, Pasquali teaches a browser user interface (fig. 1a; col. 5, line 52; *browser client*), wherein a browser executes on a client system (fig. 1c; col. 7, line 2; *client system*) to present the browser user interface on a graphical display to a user of the client system (fig. 1c; col. 7, line 2; *client system*) and accept user input from the user (fig. 5a, *step S5-4*; col. 57 line 37; *user specifies a WWW web site*), and wherein the browser (fig. 3; col. 11 line 15; *browser client*) includes functionality to request pages from servers over a network (fig. 3; col. 11 line 17; *after requesting the same via an electronic data network*) and to process received pages (fig. 3a, *step S3-3*; col. 11 lines 19-20; *constructs a web site ... based on the received content*) for presentation to the user (fig. 3a, *step S3-4*; col. 11 lines 26-27; *manifest the web site view*), the browser user interface comprising:



a page display, wherein elements of a received page are presented according to browser interpretation of data from the received page (col. 6 lines 11-12; *dynamic window in which WWW content is normally displayed*);

a rotation display area comprising some or all of a display area used for that page display (col. 9, line 56; *window modules 202 arranged in table fashion*);

storage for a plurality of rotation display items, a rotation display items storage comprising storage for a summary and a primary presentation for each rotation display item (fig. 1c, *data storage sub system 114*; col. 7 lines 7-8; *access and download HTML documents ... and other related files*);

logic for displaying, by the browser, primary presentations for less all of the plurality of rotation display items in the rotation display area (col. 8, lines 43-44; *to close module 118 via the "X" icon much like windows are closed*);

logic for displaying, by the browser, summaries for items wherein the number of summaries is greater than the number of primary presentations presented at one time (fig. 2b; *content display layer; note one primary display layer, and two summary windows*);

logic for highlighting, among the summaries displayed, the ones of the summaries that correspond to the primary presentations displayed in the rotation display area (col. 3, line 10; *may be perused for news story highlight*);

and logic for rotating, the plurality of rotation that display items to display primary presentations for a different subsets of the rotation display items (col. 15; line 25; *get all module data and order it by priority*) and for updating highlighting of

Art Unit: 2174

summaries to correspond to the different subsets of rotation display items (col. 1, lines 57-59; *select topic areas from among a pre-configured ... list of hyper-text links related to a particular filed of interest*).

As per claim 15, which is dependent on claim 14, Pasquali teaches a browser user interface wherein the number of primary presentations presented at one time is one primary presentation (fig. 2b; *content display layer; note one primary display layer, and two summary windows*).

As per claim 16, which is dependent on claim 14, Pasquali teaches a browser user interface wherein the number of summaries displayed are all of the summaries in the storage for rotation display items (fig. 2a; *note all summaries available*).

As per claim 17, which is dependent on claim 14, Pasquali teaches a browser user interface further comprising:

logic to designate an order of presentation of the rotation display items (col. 15; line 25; *get all module data and order it by priority*);

logic to modify the order of presentation based on user input such that a user indication of interest in a summary for one item results in the primary presentation for the item of interest being presented earlier in the order than if no user indication of interest was input (col. 1, lines 57-59; *select topic areas from among a pre-configured ... list of hyper-text links related to a particular filed of interest*).

As per claim 18, which is dependent on claim 14, Pasquali teaches a browser user interface wherein the logic for rotating includes logic for fading out a

current primary presentation and fading in a next primary presentation (col. 6, lines 34-37; *can manifest content based on operations occurring within a Module (e.g., ... occurrence of another event)*)).

As per claim 19, which is dependent on claim 14, Pasquali teaches a browser user interface wherein the logic for rotating includes logic for rotating out primary presentations at the differing times when multiple primary presentations are presented at any one time (col. 15, lines 17-18; *The following is all information relating to module objects. Change the order they appear in simply by changing the priority number*).

As per claim 20, which is dependent on claim 14, Pasquali teaches a browser user interface wherein the storage for the plurality of rotation display items is stored entirely within the client system (fig. 1c, *data storage sub system 114*; col. 7 lines 7-8; *access and download HTML documents ... and other related files*).

As per claim 21, which is dependent on claim 14, Pasquali teaches a browser user interface wherein the highlighting is one or more of bolding, underlining, presenting in a distinct font, presenting in a distinct color, or animating (col. 3, line 10; *may be perused for news story highlight*; also note use of <b> and <u> tags in HTML code, representing bold and highlighting).

As per claim 22, which is dependent on claim 14, Pasquali teaches a browser user interface further comprising:

a transition indicator indicating an imminent transition from a current set of one or more primary presentations to a next set of one or more primary presentations

(col. 6, lines 34-37; *can manifest content based on operations occurring within a Module (e.g., ... occurrence of another event)*); and

logic to interrupt the imminent transition upon receipt of an interrupting user input, whereby the current set of one or more primary presentations remains presented (col. 6, lines 34-37; *can manifest content based on operations occurring within a Module (e.g., ... occurrence of another event)*).

As per claim 23, which is dependent on claim 14, Pasquali teaches a browser user interface wherein the rotation display area is associated with a topic specific page, wherein topics include news, sports, weather, commentary, commerce, music, movies, games or local information (fig. 2a, *NEWS*; col. 1, line 21; *news, weather, sports*).

As per claim 24, which is dependent on claim 23, Pasquali teaches a browser user interface wherein the rotation display area is associated with a news page (fig. 2a, *NEWS*; col. 1, line 21; *news, weather, sports*) and the rotation display area displays news headlines has item the summaries and additional details related to the news headlines as the primary presentations (col. 3, line 10; *may be perused for news story highlight*).

As per claim 25, which is dependent on claim 24, Pasquali teaches a browser user interface wherein the news headlines presented as the primary presentations (col. 3, line 10; *may be perused for news story highlight*) comprise one or more of text, image, audio or video presentations (col. 8 lines 62-64; *(e.g., text, graphics, etc.) may be manifested, while ... (a news fee) ... may also be displayed*).

As per claim 26, which is dependent on claim 14, Pasquali teaches a browser user interface, the browser further comprising:

logic to track user interface interactions with the rotation display area (col. 17 lines 13-15; *GUI manipulation of window module attributes stored*); and

logic for generating messages to a server corresponding with the rotation display area, wherein a message to the server indicates user interactions (col. 17, lines 11-12; *database table is generated to store window module properties and the like*) and wherein such interactions are effected independent of whether the server receives the message (col. 18, lines 12-13; *may be downloaded ... within their WWW browser clients*).

As per claim 27, which is dependent on claim 14, Pasquali teaches a browser user interface wherein pages with received from a server include substitute presentations for presentation by browsers that do not support a rotation display area (col. 10, lines 46-48; *may be used to facilitate manifestation of content that would otherwise not be destined for window module manifestation*).

As per independent claim 28, Pasquali teaches a browser user interface (fig. 1a; col. 5, line 52; *browser client*), wherein a browser executes on a client system (fig. 1c; col. 7, line 2; *client system*) to present the browser user interface on a graphical display to a user of the client system (fig. 1c; col. 7, line 2; *client system*) and accept user input from the user (fig. 5a, step S5-4; col. 57 line 37; *user specifies a WWW web site*), and wherein the browser (fig. 3; col. 11 line 15; *browser client*) includes functionality to request pages from servers over a network (fig. 3; col. 11 line 17; *after*

*requesting the same via an electronic data network*) and to process received pages (fig. 3a, *step S3-3*; col. 11 lines 19-20; *constructs a web site ... based on the received content*) for presentation to the user (fig. 3a, *step S3-4*; col. 11 lines 26-27; *manifest the web site view*), the browser user interface comprising:

storage for a plurality of layer datasets for a received page (fig. 1c, *data storage sub system 114*; col. 7 lines 7-8; *access and download HTML documents ... and other related files*);

a page display layer, wherein elements of a main layer dataset of a received page are presented according to browser interpretation of data from the received page (col. 6 lines 11-12; *dynamic window in which WWW content is normally displayed*);

a tool layer having elements related to an activity (col. 9, lines 56-57; *web site (environment) controls*);

logic to optionally display the tool layer over the page display layer (fig. 2a and 2b, *note control tool layer over display layer 202, 208*); and

logic to accept input from the user related to the activity and removing the tool layer display when complete (col. 8, lines 43-44; *to close module 118 via the "X" icon much like windows are closed*).

As per claim 29, which is dependent on claim 28, Pasquali teaches a browser user interface wherein the two layer is semi-transparent (col. 33, line 30; *spacer graphic ie. a transparent image that is set by the width & height values specified*).

As per claim 30, which is dependent on claim 28, Pasquali teaches a browser user interface, the browser further comprising:

logic to track user interface interactions with the tool layer (col. 17, lines 13-15; *GUI manipulation of window module attributes stored*); and

logic for generating messages to a server corresponding with the tool layer, wherein a message to the server indicates user interactions (col. 17, lines 11-12; *database table is generated to store window module properties and the like*) and wherein such interactions are effected independent of whether the server receives the message (col. 18, lines 12-13; *may be downloaded ... within their WWW browser clients*).

As per claim 31, which is dependent on claim 28, Pasquali teaches a browser user interface wherein pages with received from a server include substitute presentations for presentation by browsers that do not support multiple layers (col. 10, lines 46-48; *may be used to facilitate manifestation of content that would otherwise not be destined for window module manifestation*).

As per independent claim 32, Pasquali teaches a browser user interface (fig. 1a; col. 5, line 52; *browser client*), wherein a browser executes on a client system (fig. 1c; col. 7, line 2; *client system*) to present the browser user interface on a graphical display to a user of the client system (fig. 1c; col. 7, line 2; *client system*) and accept user input from the user (fig. 5a, step S5-4; col. 57 line 37; *user specifies a WWW web site*), and wherein the browser (fig. 3; col. 11 line 15; *browser client*) includes functionality to request pages from servers over a network (fig. 3; col. 11 line 17; *after*

*requesting the same via an electronic data network*) and to process received pages (fig. 3a, *step S3-3*; col. 11 lines 19-20; *constructs a web site ... based on the received content*) for presentation to the user (fig. 3a, *step S3-4*; col. 11 lines 26-27; *manifest the web site view*), the browser user interface comprising:

storage, at the client system, for a plurality of page components, wherein a page received from a server is displayable as a collection of page components (fig. 1c, *data storage sub system 114*; col. 7 lines 7-8; *access and download HTML documents ... and other related files*) each having a display position (col. 11, lines 66-67; *defines a screen position for a window module*);

logic for accepting user input for modifying page component display positions (col. 12, lines 64-65; *permit user adjustment of window module attributes (e.g., vertical and horizontal CME positions ...)*); and

logic for revising a page display according to user input for modifying page component display positions (col. 12, lines 64-65; *permit user adjustment of window module attributes (e.g., vertical and horizontal CME positions ...)*) without requiring further interactions with the server (col.4 lines 36-41; *window object(s) ... may be updated and loaded with content ... without requiring the content manifestation environment to be refreshed*).

As per claim 33, which is dependent on claim 32, Pasquali teaches a browser user interface further comprising logic for generating a message from the client system to the server representing modified page component display positions to allow subsequent presentations of pages from the server to be modified according to the



modified page component display positions (col. 52, lines 12-13; *positions it in the proper place based on the variables set*).

As per independent claim 34, Pasquali teaches a method of user interaction (fig. 5a) with a browser user interface (fig. 1a; col. 5, line 52; *browser client*), wherein a browser executes on a client system coupled to a server (fig. 1a; col. 7, lines 18-19; *may be coupled with SVR system 102*) over a network (fig. 1a; col. 7 line 19; *via the electronic data network*) and provides a browser user interface (fig. 3; col. 11 line 15; *browser client*) to server resources (fig. 3, step S3-2; col. 11 lines 15-16; *all files and functions and content stored with in [sic] server side systems*) and wherein the browser includes functionality to request pages from servers over the network (fig. 3; col. 11 line 17; *after requesting the same via an electronic data network*) and to process received pages (fig. 3a, step S3-3; col. 11 lines 19-20; *constructs a web site ... based on the received content*) for presentation to a browser user (fig. 3a, step S3-4; col. 11 lines 26-27; *manifest the web site view*), the browser user interface comprising:

requesting a page, using the browser, from a target server (fig. 5a, step S5-4; col. 57 line 37; *user specifies a WWW web site*);

receiving the requested page at the client system from the target server (fig. 5a, step S5-5; col. 57 lines 43-44; *receives window content*), wherein the received requested page comprises a plurality of layers (col. 51, line 3; *window modules and layers*; see also col. 5, line 64-col. 6, line 8 for definition of layers), where at least one of the plurality of layers is a page display layer and at least one of the plurality of layers other than the page display layer is an optional display layer (fig. 2b; *content display*

*layer, note one primary display layer, and two summary display layers) comprising at least one dynamic interface element corresponding to possible user input (col. 6, lines 1-3; may be scaled, dragged, or otherwise operated upon);*

generating a user display corresponding to the received requested page (fig. 5a, step S5-6; col. 57, line 47; *causes display of received window content, HTML*);

displaying the user display and accepting user input corresponding to the user display of the received requested page (col. 3, lines 30-34 *user-selectable screen-related operation/event occurs within a web browser screen (e.g., such as ... the occurrence of an appropriate mouse-click event)*);

when a user input corresponding to a request for display of the optional display layer, modifying the display to present the optional display layer (col. 6, lines 1-4; *may be scaled, dragged, or otherwise operated upon ... may be associated with program logic*);

when the optional display layer is presented, accepting user input corresponding to dynamic interface elements of the optional display layer (col. 6, lines 1-4; *may be scaled, dragged, or otherwise operated upon ... may be associated with program logic*);

storage for recording the accepted user input (fig. 1c, data storage sub system 114; col. 7 lines 7-8; access and download HTML documents ... and other related files); and

taking an action corresponding to the recorded accepted user input (col. 6, lines 1-4; *may be scaled, dragged, or otherwise operated upon ... may be associated with program logic*).

As per claim 35, which is dependent on claim 34, Pasquali teaches a method wherein the plurality of layers comprises one or more of a slide sheet (fig. 2a; *content display section 242*; col. 9 line 62; *may either be minimized or maximized*), a rotation display area (col. 1, lines 57-59; *select topic areas from among a pre-configured ... list of hyper-text links related to a particular field of interest*), a tool layer with user inputs (col. 9, lines 56-57; *web site (environment) controls 210*), and a menu bar (col. 51, lines 8-11; *initiates a pop-up window ... "popMenu()"*).

As per claim 36, which is dependent on claim 34, Pasquali teaches a method wherein the plurality of layers is structured within the received requested page according to DHTML (col. 6, lines 6-7; *HTML rendition model such as those defined by DHTML standards*).

As per claim 37, which is dependent on claim 34, Pasquali teaches a method wherein modifying the display to present the optional display layer comprises semi-transparently overlaying the page display layer with the optional display layer (col. 33, line 30; *spacer graphic ie. a transparent image that is set by the width & height values specified*).

As per claim 38, which is dependent on claim 34, Pasquali teaches a method further comprising a step of transmitting the recorded accepted user input asynchronously to the target server (col. 17 lines 11-16; *module attributes stored within*

*the generated database table which is stored at the server side; also note col. 7 lines 54-58; Any computing system that facilitates service of web related documents).*

As per claim 39, which is dependent on claim 34, Pasquali teaches a method further comprising a step of transmitting the recorded accepted user input asynchronously to a recording server referenced in the received requested page (col. 17 lines 11-16; *module attributes stored within the generated database table which is stored at the server side; also note col. 7 lines 54-58; Any computing system that facilitates service of web related documents).*

As per claim 40, which is dependent on claim 34, Pasquali teaches a method wherein the optional display layer comprises at least one of a tool layer, an e-mail entry layer, a calendar entry layer, a photo review layer, a news layer, an instant messaging layer, and a voice chat layer (fig. 2a; *tool layer 210, EMAIL, NEWS, CHAT).*

As per claim 41, which is dependent on claim 34, Pasquali teaches a method wherein the dynamic interface elements include at least one of one or more button (col. 2 line 20; *navigation buttons*), one or more entry field or one or more form (col. 6 line 3; *form element*).

### **Conclusion**

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Bladow et al. (US #6,115,040) teaches a graphical user interface for web enabled applications including frames, toolbars, layers, etc.

***Inquiries***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chris Watt whose telephone number is **(571) 270-1046**. The examiner can normally be reached on Monday-Thursday 6:30-4:00 Eastern. The examiner can also be reached on alternate Fridays.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kristine L. Kincaid can be reached on (571) 276-5619. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

\Chris Watt\

October 5, 2006

CAW

*Kristine Kincaid*  
KRISTINE KINCAID  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100